

日 本 国 特 許 庁

PATENT OFFICE  
JAPANESE GOVERNMENT

Jc525 U.S. PRO  
09/506650

別紙添付の書類に記載されている事項は下記の出願書類に記載されて  
る事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed  
with this Office.

出 願 年 月 日  
Date of Application:

1999年 2月18日

願 番 号  
Application Number:

平成11年特許願第040336号

願 人  
Applicant(s):

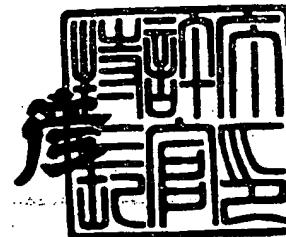
ソニー株式会社

CERTIFIED COPY OF  
PRIORITY DOCUMENT

1999年12月10日

特許庁長官  
Commissioner,  
Patent Office

近 藤 隆 彦



【書類名】 特許願

【整理番号】 9900120202

【提出日】 平成11年 2月18日

【あて先】 特許庁長官殿

【国際特許分類】 H04L 12/00

【発明者】

    【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社  
内

    【氏名】 原 和弘

【発明者】

    【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社  
内

    【氏名】 藤井 昇

【特許出願人】

    【識別番号】 000002185

    【氏名又は名称】 ソニー株式会社

    【代表者】 出井 伸之

【代理人】

    【識別番号】 100080883

    【弁理士】

    【氏名又は名称】 松隈 秀盛

    【電話番号】 03-3343-5821

【手数料の表示】

    【予納台帳番号】 012645

    【納付金額】 21,000円

【提出物件の目録】

    【物件名】 明細書 1

    【物件名】 図面 1

    【物件名】 要約書 1

特平 1 1 - 0 4 0 3 3 6

【包括委任状番号】 9707386

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 通信方法及び通信装置

【特許請求の範囲】

【請求項 1】 片方向の第 1 の通信回線にデータを送出するブリッジタイプの送出手段に、双方向通信が可能な第 2 の通信回線を接続して第 1 の通信回線で仮想的に双方向通信を行う通信方法において、

上記送信手段に所定のインターフェースから入力されたパケットの宛先を判断し、その判断したパケットの宛先からそのパケットかどのインターフェースの先にあるネットワークに送られるべきかを判断し、転送が必要な場合のみに所定のインターフェースから転送を行う

通信方法。

【請求項 2】 請求項 1 に記載した通信方法において、

送信側のネットワークに接続されているノードのアドレスを自動的に上記送信手段が検出することを特徴とする

通信方法。

【請求項 3】 請求項 2 に記載した通信方法において、

自動的に検出した送信側のネットワークに接続されているノードのアドレスをリストにして上記送信手段が保持し、そのリストを基にしてパケットの転送の判断を行うことを特徴とした

通信方法。

【請求項 4】 請求項 3 に記載した通信方法において、

自動的に検出した送信側のネットワークに接続されているノードのアドレスのリストを上記送信手段が定期的に更新し、一定時間以上の間パケットのやり取りが行われていないノードのアドレスをリストから削除することを特徴とした

通信方法。

【請求項 5】 片方向の第 1 の通信回線にデータを送出するブリッジタイプの送出手段として構成される通信装置において、

双方向通信が可能な第 2 の通信回線を接続するインターフェースを備えると共に、

所定のインターフェースから入力されたパケットの宛先を判断し、その宛先からそのパケットがどのインターフェースの先にあるネットワークに送られるかを判断し、転送が必要であると判断したときのみ転送処理を実行させる制御手段を備えた

通信装置。

【請求項 6】 請求項 5 に記載した通信装置において、

上記制御手段は、インターフェースに接続されたネットワークに接続されているノードのアドレスを自動的に検出する検出手段を備えたことを特徴とする

通信装置。

【請求項 7】 請求項 6 に記載した通信装置において、

上記検出手段が自動的に検出したノードのアドレスをリストにして保持するアドレス記憶手段を備え、

上記制御手段は、上記アドレス記憶手段に記憶されたリストを基にしてパケットの転送の判断を行うことを特徴とした

通信装置。

【請求項 8】 請求項 7 に記載した通信装置において、

上記制御手段は、上記アドレス記憶手段に記憶されたリストを定期的に更新し、一定時間以上の間パケットのやり取りが行われていないノードのアドレスをリストから削除することを特徴とした

通信装置。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、例えば衛星回線などの片方向の通信回線を使用して、インターネットの通信を行う通信方法及びその通信方法に適用される通信装置に関する。

【0 0 0 2】

【従来技術】

ある通信回線を流れる I P データグラムを、衛星回線のような片方向の通信回線に対して転送する通信機器（送出機）を、フィード（F e e d）と呼ぶ。フィ

ードには、ルータとしての実装と、ブリッジとしての実装が考えられている。また、フィールドには、片方向の通信回線を双方向の通信回線に見せかけるための技術であるUDLR (Uni-Directional Link Routing) を実装しているものがある (UDLRについては、Internet-Draft: draft-ietf-udlr-lltunnel-01.txtを参照)。以下の説明で示されるフィールドは、UDLRをサポートした、ブリッジタイプのフィールドである。

## 【0003】

UDLRとしての双方向通信をサポートした、ブリッジタイプのフィールドの構成を図12に示す。フィールド1は少なくとも3つのネットワークインターフェースを持つ。それらのインターフェースを、ここではローカルI/F2, UDL I/F3, トンネルI/F4と呼ぶことにする。ローカルI/F2は、双方向のネットワークインターフェースで、このI/Fが接続されているネットワークを、ローカルネットワーク5と呼ぶ。UDL I/F3は、送信専用の片方向のネットワークインターフェースで、このI/Fが接続されているネットワークを、UDLネットワーク6と呼ぶ。トンネルI/F4は、双方向のネットワークインターフェースで、このI/Fが接続されているネットワークを、トンネル (Tunnel) ネットワーク7と呼ぶ。ここで、ローカルネットワーク5と、UDLネットワーク6は、同じネットワークアドレス8を持つ。また、トンネルネットワーク7は、ローカルネットワーク5や、UDLネットワーク6とは、異なるネットワークアドレス9を持つ。

## 【0004】

フィールド1の主要な機能は、図13のフローチャートに示すように、ローカルネットワーク5を流れるパケットを全て受信し (S201)、そのパケットをUDL I/F3からUDLネットワーク6へ転送することである (S202)。これをパケット転送機能10と呼ぶ。

## 【0005】

またフィールド1の他の機能として、UDLRとしての双方向通信を実現するために、図15に示すように、トンネルI/F4からGREパケット11 (RFC

1701参照)と呼ばれる、他のパケット12をIPデータグラム内にカプセル化したパケットを受信し(S401)、GREパケット11がフラグメントされているかどうかを調べる(S402)、もしGREパケット11がフラグメント化されていたら再生構築の処理を行い(S403)、その後にGREパケット11にカプセル化されていたパケット12を取り出し(S404)、その取り出したパケット12のコピーを2つ作成し(S405)、1つをローカルI/F2からローカルネットワーク5へ、もう1つをUDL I/F3からLDLネットワーク6へと送出する(S406)という機能がある。これをUDLR機能13と呼ぶ。このUDLR機能13が実行される構成を図14に示す。

【0006】

【発明が解決しようとする課題】

フィードの他の機器との接続形態の一例を図16に示す。ローカルネットワーク5には、フィード1以外に、ルータ14や、ホスト15が繋がっている。UDLネットワーク6には、受信専用のインターフェース16を持つ受信装置であるレシーバ17が繋がっている。トンネルネットワーク7には、ルータ18が繋がっている。ルータ18はルータ14と同一のルータでも構わないが、その場合はローカルネットワーク5とトンネルネットワーク7とは異なるインターフェースを用いて接続されているものとする。従来では、図17に示すように、ローカルネットワーク5を経由してルータ14とホスト15の間の通信を行うと、その送信により伝送されるパケット19は、フィード1のパケット転送機能10により、UDLネットワーク6にも伝送される。ここでUDLネットワーク6に転送されたパケットをパケット20とする。しかし、パケット20は、受け取る機器が存在しないパケットである。このようなパケットがUDLネットワーク6を流れることは、UDLネットワーク6の帯域を無駄遣いすることになる。また、UDLネットワーク6を接続されたレシーバ17のような機器では、この不要なパケット20をフィルタリングによって破棄するといった作業を行うことになり、CPUなどの資源を無駄に使うことになってしまう。

【0007】

また、図18に示すように、トンネルネットワーク7から受け取ったGREパ

ケット 11 にカプセル化されていたパケット 12 が、ルータ 14 やホスト 15 などのローカルネットワーク 5 に接続されている機器宛のものであった場合も、フィード 1 の UDLR 機器 13 により、パケット 12 のコピーが LDL ネットワーク 6 に伝送されてしまう。この UDL ネットワーク 6 を流れるパケット 12 のコピーを、パケット 21 とする。パケット 21 もパケット 20 と同様に受け取る機器が存在しないパケットであり、このようなパケットが UDL ネットワーク 6 を流れることは、UDL ネットワーク 6 の帯域を無駄遣いすることになる。また、UDL ネットワーク 6 に接続されたレシーバ 17 のような機器では、この不要なパケット 21 をフィルタリングによって破棄するといった作業を行うことになり、CPU などの資源を無駄に使うことになってしまう。

## 【0008】

本発明の目的は、上述したパケット 20 やパケット 21 のような不要なパケットを、フィードができるだけ生成しないようにし、UDL ネットワークの帯域を有効に利用できるようにし、また、UDL ネットワークに接続されたレシーバの様な機器の負荷を軽減することにある。

## 【0009】

## 【課題を解決するための手段】

本発明は、片方向の第 1 の通信回線にデータを送出するブリッジタイプの送出手段に、双方向通信が可能な第 2 の通信回線を接続して第 1 の通信回線で仮想的に双方向通信を行う場合に、送信手段に所定のインターフェースから入力されたパケットの宛先を判断し、その判断したパケットの宛先からそのパケットかどのインターフェースの先にあるネットワークに送られるべきかを判断し、転送が必要な場合のみに所定のインターフェースから転送を行うようにしたものである。

## 【0010】

本発明によると、UDLR のような双方向通信をサポートしたブリッジタイプのフィードが、そのフィードに繋がる双方向の通信回線を流れるパケットのうち、片方向の通信回線へ転送する必要のないパケットを、片方向の通信回線へ転送してしまう回数を減らすことができる。

## 【0011】



## 【発明の実施の形態】

以下に、本発明の実施の形態について説明する。

本実施の形態による基本的な処理は、ローカルネットワーク 5 に接続されたノード（ルータ 14 やホスト 15 などの通信機器）のアドレスをフィード 1 が記憶しておくことで、ローカルネットワーク 5 に接続されたノード宛のパケット UDL ネットワーク 6 に転送しないようにするというものである。

## 【0012】

これを実現するために、まず、フィード 1 がローカルネットワーク 5 に接続されたノードのアドレスを学習する処理について説明する。次に、その学習したアドレスを利用して、フィード 1 がどのように UDL ネットワーク 6 に無駄なパケットを転送しないようにするかを説明する。続いて、フィード 1 が学習したアドレスを自動的に更新する方法について説明し、最後に、実装上の工夫について説明する。説明をする上で、ローカルネットワーク 5 やトンネルネットワーク 7 は、10baseT や 100baseTX などのイーサネット（Ethernet）であることを想定して説明を進めていくが、イーサネットでなくても、データフォーマットにおいて宛先アドレスと送信元アドレスを持つ処理ならばどのようなものでも構わない。

## 【0013】

では、フィード 1 がローカルネットワーク 5 に接続されたノードのアドレスを学習する処理について説明するが、基本的なアイデアは、フィード 1 はパケット転送機能 10 のためにローカルネットワーク 5 を流れるパケットを全て取り込むのだから、その際に取り込んだパケットの送信アドレスを学習するという処理である。これを図 1 のフローチャートを用いて説明する。

## 【0014】

まず、フィード 1 は、ローカルネットワーク 5 に流れるパケットを監視している（S801）。そこで、ローカルネットワーク 5 に接続されたノードが、パケット 22 を送出する（S802）。このパケット 22 は、IP データグラムや、ICMP、ARP などの任意のパケットで、いずれもイーサネットフレームの状態でローカルネットワーク 5 を流れている。また、パケット 22 の宛先は任意であ

り、ローカルネットワーク 5 のノードに限らず、他のネットワーク上のノードでも良いし、マルチキャストやブロードキャストでも構わない。

【0015】

すると、フィールド 1 は、ローカル I/F 2 からこのパケット 22 をフィールド 1 の内部に取り込む (S803)。フィールド 1 のローカル I/F 2 は、パケット転送機能 10 の実現のために、自分宛のパケットに限らずローカルネットワーク 5 を流れるパケットは全て取得できるようになっているので、パケット 22 を取り込むことが可能なのである。

【0016】

次に、フィールド 1 は、パケット 22 からイーサネットフレームの送信元アドレス 23 を取り出す (S804)。パケット 22 のデータフォーマットは、例えば図 2 に示すように構成される。そして、フィールド 1 はローカルネットワーク 5 に接続されたノードのアドレスのリスト 24 からパケット 22 の送信元アドレス 23 を検索する (ステップ S805)。リスト 24 の構造は、例えば図 3 に示すようになっており、各行は、ローカルネットワーク 5 に接続されたノードのアドレスを入れる列 25 と、そのアドレスを最後に学習した時間を入れる列 26 から構成されている。

【0017】

フィールド 1 は、リスト 24 のアドレスの列 25 にアドレス 23 が存在するかどうかを調べ (S806)、存在する場合は、そのアドレスが存在する行の、時間を入れる列 26 の値を、現在の時間に更新する (S807)。一方、リスト 24 のアドレスの列 25 にアドレス 23 が存在しない場合は、新たな行を作成し、その列、25 にアドレス 23 を入れ、列 26 には、現在の時間を入れる (S808)。

【0018】

これでパケット 22 からのアドレスの学習を終え、フィールド 1 は、ステップ S801 に戻って、ローカルネットワーク 5 を流れる次のパケットからのアドレスの学習に備える。以上の方法により、フィールド 1 はローカルネットワーク 5 に接続されたノードのアドレスを学習する。

## 【0019】

では次に、学習したアドレスを利用して、フィールド1がどのようにUDLネットワーク6に無駄なパケットを転送しないようにするかを説明する。これは、フィールド1の機能毎にフローチャートを用いて説明する。最初に、パケット転送機能10の場合について説明し、次に、UDLR機能13の場合について説明する。

## 【0020】

まず、フィールド1がパケット転送機能10を行う際には、図4のフローチャートに示すようにして無駄なパケットの転送を防ぐ。最初は図1の処理と同じである。まず、フィールド1は、ローカルネットワーク5に流れるパケットを監視している(S1101)。そこで、ローカルネットワーク5接続されたノードが、パケット22を送出する(S1102)。このパケット22は、IPデータグラムや、ICMP、ARPなどの任意のパケットで、いずれもイーサネットフレームの状態でローカルネットワーク5を流れている。また、パケット22の宛先は任意であり、ローカルネットワーク5上のノードに限らず、他のネットワークのノードでも良いし、マルチキャストやブロードキャストでも構わない。

## 【0021】

すると、フィールド1は、ローカルI/F2からこのパケット22をフィールド1の内部に取り込む(S1103)。次に、フィールド1は、パケット22からイーサネットフレームの宛先アドレス27を取り出す(S1104)。そして、リスト24のアドレスの列25に取り出した宛先アドレス27が存在するかを検索する(S1105)。フィールド1は、検索の結果、宛先アドレス27がリスト24に存在するかどうかでパケット22を転送するかどうかを決める(S1106)。存在した場合は、そのパケットはローカルネットワーク5上のノード宛のものであり、UDLネットワーク6に転送する必要がないので、破棄する(S1107)。一方、リスト24のアドレスの列25に取り出した宛先アドレス27が存在しない場合は、そのパケットはローカルネットワーク5上には存在しないノード宛のものであるとみなして、UDLネットワーク6へ転送する(S1108)。

## 【0022】

これでパケット転送機能10におけるパケット22の処理を終え、フィード1は、S1101に戻って、ローカルネットワーク5を流れる次のパケットの処理に備える。以上の処理により、フィード1ネットワーク5にながれるパケットを無駄にUDLネットワーク6に転送するのを防ぐ。

## 【0023】

次に、UDLR機能13の場合について説明する。フィード1がUDLR機能13を行う際には、図5のフローチャートに示すように処理して無駄なパケットの転送を防ぐ。まず、フィード1は、トンネルネットワーク7に流れるパケットを監視している(S1201)。そこで、トンネルネットワーク7接続されたノードが、GREパケット11を送出する(S1202)。このGREパケット11には、図6に示すように、IPデータグラムや、ICMP、ARPなどの任意のパケット12がイーサネットフレームの形でIPデータグラムのデータ部分にカプセル化されており、そのIPデータグラムがさらにイーサネットフレームに入った状態でトンネルネットワーク7を流れている。フィード1は、このGREパケット11のして先IPアドレス28がフィード1のトンネルI/F4のIPアドレスであり、かつIPヘッダ29のプロトコルフィールド30の値が所定値(ここではGERパケットであることを示す値である47)であるかを調べ(S1203)、そうである場合は、次のステップS1204に進み、そうでない場合は、UDLRに関係ない通常のパケットとみなして、通常のパケット用の処理S1205を行い、その後待機状態のステップS1201に戻る。

## 【0024】

ステップS1204では、フィード1はGREパケット11がフラグメント化されているかどうかを調べる。フラグメント化されている場合は、再構築の処理(S1206)を行うのだが、ここでは再構築の処理に関する詳細な説明は省き、フィード1はステップS1206からステップS1207に進んで、再構築の処理が終わった完全なGREパケット31を取り込むものとする。GREパケット11がフラグメント化されていない場合は、GREパケット11をそのまま完全なGREパケット31としてフィード1に取り込み、ステップS1207に進

む。

【0025】

次に、フィールド1は、GREパケット31からイーサネットフレーム32を取り出す(S1207)。さらに、そのイーサネットフレーム32から、イーサネットの宛先アドレス33を取り出す(S1208)。そして、リスト24のアドレスの列25に取り出した宛先アドレス33が存在するかを検索する(S1209)。存在した場合は、そのパケットはローカルネットワーク5上のノード宛のものであり、UDLネットワーク6に転送する必要がないので、イーサネットフレーム32をローカルI/Fから出力し、ローカルネットワーク上へと転送する(S1210)。一方、リスト24のアドレスの列25に取り出した宛先アドレス33が存在しない場合は、そのイーサネットフレーム32はローカルネットワーク5上には存在しないノード宛ものが、ローカルネットワーク5上にあるノードだがまだそのアドレスをフィールド1が学習していないノード宛のものであるか、またはブロードキャストやマルチキャストであるとみなして、そのイーサネットフレーム32のコピーを2つ作成し、1つをローカルI/F2からローカルネットワーク5へ、もう1つをUDL I/F3からUDLネットワーク6へと転送する(S1211)。

【0026】

これでUDLR機能13におけるGREパケット11の処理を終え、フィールド1は、ステップS1201に戻って、トンネルネットワーク7を流れる次のパケットの処理に備える。以上の方法により、フィールド1はトンネルネットワーク5から入力されたGREパケットの中身が無駄にUDLネットワーク6に転送するのを防ぐ。

【0027】

では、次に、フィールド1が学習したアドレスを自動的に更新する処理について説明する。フィールド1が、学習したアドレスを削除することなく永遠に保持しつづけると、問題が発生することがある。それは、図7に示すように、ローカルネットワーク5に接続されたノード34が、ローカルネットワーク5から取り外されて、UDLネットワーク6など他のネットワークに接続された場合に発生する

。問題の詳細は次のようになっている。まず、ノード34がローカルエリアネットワーク5に接続されている状態でフィールド1がノード34のアドレスを学習する。次に、ノード34をローカルネットワーク5から取り外し、UDLネットワーク6の受信側に接続する。その状態でローカルネットワーク5上のノード35からUDLネットワーク6の上ノード34にパケットを送ろうとしても、ノード34のアドレスがリスト24に残っているために、フィールド1はそのパケットを破棄してしまう。そのため、パケットはUDLネットワーク6には転送されず、従ってノード35からノード34への通信を行うことはでなくなってしまう。

## 【0028】

この問題を解決するために、フィールド1は、リスト24にあるアドレスを永遠に保持するのではなく、定期的にリスト24を更新し、不要なアドレスを削除するようにする。削除を行う場合には、例えば一定時間以上ローカルネットワーク5にパケットを送出しないノードのアドレスを削除するものとする。一定時間というのは、ローカルネットワーク5上のノード34がローカルネットワーク5から取り外され、他のネットワークに繋がるのにかかる時間より短い時間を目安とする。具体的には、例えば3分程度にしておけば問題ないと思われる。

## 【0029】

削除の流れを図8のフローチャートを参照して説明する。フィールド1は、タイマを用いて、一定時間毎にリスト24の全ての行を検索する。まずタイマから通知があるまで待機し（S1501）、通知が来て検索の時間が訪れると（S1502）、リスト24の最初の最初の行を取り出し（S1503）、その行の時間が入った列26の値を読み込む（S1504）。この値（時間）を現在の時間から引き、その差があらかじめ指定された時間（例えば3分）より大きいかどうかを比較する（S1505）。大きい場合は、リスト24のこの行のアドレスが示すノードは、指定された時間以上の間、ローカルネットワーク5にパケットを送出していないことになるので、この行を削除する（S1506）。差が指定された時間より大きくない場合は、この行はそのままにしておく。いずれの場合も、この処理を次の行以降も繰り返す。つまり、次の行が存在するかを調べ（S1507）、存在すればその行を読み込む（S1508）、ステップS1504に戻

る。次の行が存在しなければ、そこでまた待機状態（ステップ S 1 5 0 1）に入る。

【0 0 3 0】

以上が、フィード 1 が学習したアドレスを自動的に更新する処理の説明である。

【0 0 3 1】

では、最後に、実装上の処理と、その他の注意点について説明する。

【0 0 3 2】

まず、フィード 1 がローカルネットワークを流れるフィード 1 自身宛のパケット 3 6 をどう処理するかについてである。パケット 3 6 がローカルネットワーク 5 を流れてきた場合は、これは UDL ネットワークに転送すべきものではないので、図 4 のステップ S 1 1 0 5 の段階で、パケット 3 6 のイーサネットフレームの宛先アドレス時自分宛であることを検出し、UDL ネットワーク 6 に転送せずに、自分自身が取り込めば良い。また、実装上の工夫として、フィード 1 は、ローカル I / F 2 以外からも自分宛のパケットを受け取ることが可能なので、実装を簡単にするため、ローカル I / F 2 からは自分宛のパケット 3 6 を受け取らずに破棄してしまうものとする。リスト 2 4 にフィード 1 自身のローカル I / F 2 のイーサネットアドレスを最初から入れておき、さらにそのアドレスの行をフィード 1 が定期的にリスト 2 4 を更新する際の対象外にしておくと、図 4 のステップ S 1 1 0 5 からステップ S 1 1 0 6 に処理が移り、自動的にパケット 3 6 は破棄される。

【0 0 3 3】

一方、パケット 3 6 が、GER パケット 1 1 にカプセル化されてトンネル I / F 4 から入ってきた場合は、これはローカルネットワーク 5 にも UDL ネットワーク 6 にも転送すべきではないので、図 5 のステップ S 1 2 0 8 の段階で、パケット 3 6 のイーサネットフレームの宛先アドレスが自分宛であることを検出し、他のネットワークには転送せずに、自分自身が取り込めば良い。この場合も、実装を簡単にするために、トンネル I / F 4 からは自分自身宛の GRE にカプセル化されたパケット 3 6 を受け取らずに破棄してしまう実装も考えられるが、この

場合は、リスト 24 にフィールド 1 自身のイーサネットアドレスを入れておくと、図 4 のステップ S 1208 からステップ S 1209 に移ってそのパケット 36 を破棄せずにローカルネットワークに転送してしまう。従って、単純にリスト 24 にフィールド 1 自身のイーサネットアドレスを入れておくだけでは対応できないので、注意が必要である。

【0034】

次に、リスト 24 の実装処理について説明する。リスト 24 を実装する際には、フィールド 1 はハードウェアまたはソフトウェアの制約により、リスト 24 を有限の長さにしか持てない。また、リスト 24 が長くなると、検索に時間がかかるようになり、高速に大量のパケットを転送するのが難しくなる。そこで、リスト 24 の長さを有限にし、検索にかかる時間を短くする。また、リスト 24 の長さが有限になると、リスト 24 の全ての行が埋まってしまった際に、新しくアドレスを追加することができなくなるので、その場合は、リスト 24 に入っているアドレスの中から、最も更新時間が古いアドレスを削除して、空いた行に新しいアドレスを追加するようにする。

【0035】

具体的には、図 9 に示すように、検索にかかる時間が問題にならない程度の長さが固定長のリスト 37 を持っておく。リスト 37 には、アドレスが入る列 25 と更新時間が入る列 26 の他に、各行が有効な行かどうかを示す列 38 がある。リスト 37 に新しいアドレスを加える場合は、図 10 に示す手順で行う。まず、リスト 37 の列 38 を調べ、無効な行があるかを調べる (S 1701)。無効な行があった場合は、その行に新しいアドレスと更新時間を入れ、その行の列 38 を有効に変更する (S 1702)。無効な行がなかった場合は、リスト 37 の列 26 を調べ、最も更新時間の古い行を調べる (S 1703)。そしてその行に新しいアドレスと更新時間を入れる (S 1704)。これで追加は完了である。

【0036】

このデータ構造でアドレスの検索を行う場合は、列 38 が有効で、かつ列 25 に該当するアドレスが入っている行を探せばよい。また、アドレスの削除を行う場合は、削除したいアドレスを前記の手順で検索し、その行の列 38 を有効から



無効に変更するだけで良い。更新時間の変更をする場合は、前記の手順で更新したアドレスを検索し、その行の列 26 を更新すれば良い。

#### 【0037】

以上、フィード 1 が UDL ネットワーク 6 に無駄なパケットを流さないための方法について説明したが、具体的なフィード 1 の使用例としては、例えば図 11 のようになる。UDL ネットワーク 6 としては、衛星回線 39 を使うことが考えられる。衛星回線 39 は大容量であるが、高価であるため、本実施の形態のように無駄なパケットを流さないようにするのは有効な技術である。ルータ 40 は、図 16 に示したルータ 14 とルータ 18 を同一のルータにまとめたものである。ここでは、レシーバ 17 はルータとして描かれているが、レシーバ 17 をブリッジとしても、フィード 1 の機能は全く問題なく動作する。このようなシステム以外でも、UDL ネットワーク 6 としてケーブルを利用した場合や、ローカルネットワーク 5 やトンネルネットワーク 7 としてイーサネットではないネットワークを利用した場合も、本発明は有効に作用する。

#### 【0038】

##### 【発明の効果】

本発明により、UDLR のような双方向通信をサポートしたブリッジタイプのフィードが、フィードに繋がる双方向の通信回線を流れるパケットのうち、片方向の通信回線へ転送する必要のないパケットを、片方向の通信回線へ転送してしまう回数を減らすことができる。

#### 【0039】

これにより、片方向の通信回線の帯域を有効にできるようにし、また、片方向の通信回線に接続されたレシーバの様な通信機器の負荷を軽減することができる。

##### 【図面の簡単な説明】

##### 【図 1】

本発明の実施の形態によるアドレス学習処理例を示すフローチャートである。

##### 【図 2】

本発明の実施の形態によるフォーマットの例を示す説明図である。

【図 3】

本発明の実施の形態によるアドレスのリストの例を示す説明図である。

【図 4】

本発明の実施の形態によるパケット転送機能の流れを示すフローチャートである。

【図 5】

本発明の実施の形態による UDL R 機能の流れを示すフローチャートである。

【図 6】

本発明の実施の形態による G E R パケットのフォーマットを示す説明図である。

【図 7】

本発明の実施の形態による構成の例（学習したアドレスを削除する必要がある例）を示すブロック図である。

【図 8】

本発明の実施の形態による学習したアドレスを削除する処理を示すフローチャートである。

【図 9】

本発明の実施の形態によるローカルネットワーク上のノードが持つアドレスのリストの実装例を示す説明図である。

【図 1 0】

本発明の実施の形態によるリストへの新しいアドレスの追加処理の例を示すフローチャートである。

【図 1 1】

本発明の実施の形態を衛星回線とインターネット（又はイントラネット）を用いたシステムへの適用例を示すブロック図である。

【図 1 2】

従来の UDL R をサポートしたブリッジタイプのフィードの構成の例を示すブロック図である。

【図 1 3】

従来のパケット転送機能の例を示すフローチャートである。

【図 14】

従来のUDLR機能を説明するためのブロック図である。

【図 15】

従来のUDLR機能の流れを示すフローチャートである。

【図 16】

従来のフィードと他の機器との接続形態の例を示すブロック図である。

【図 17】

従来のフィードにおけるパケット転送機能を説明するためのブロック図である。

【図 18】

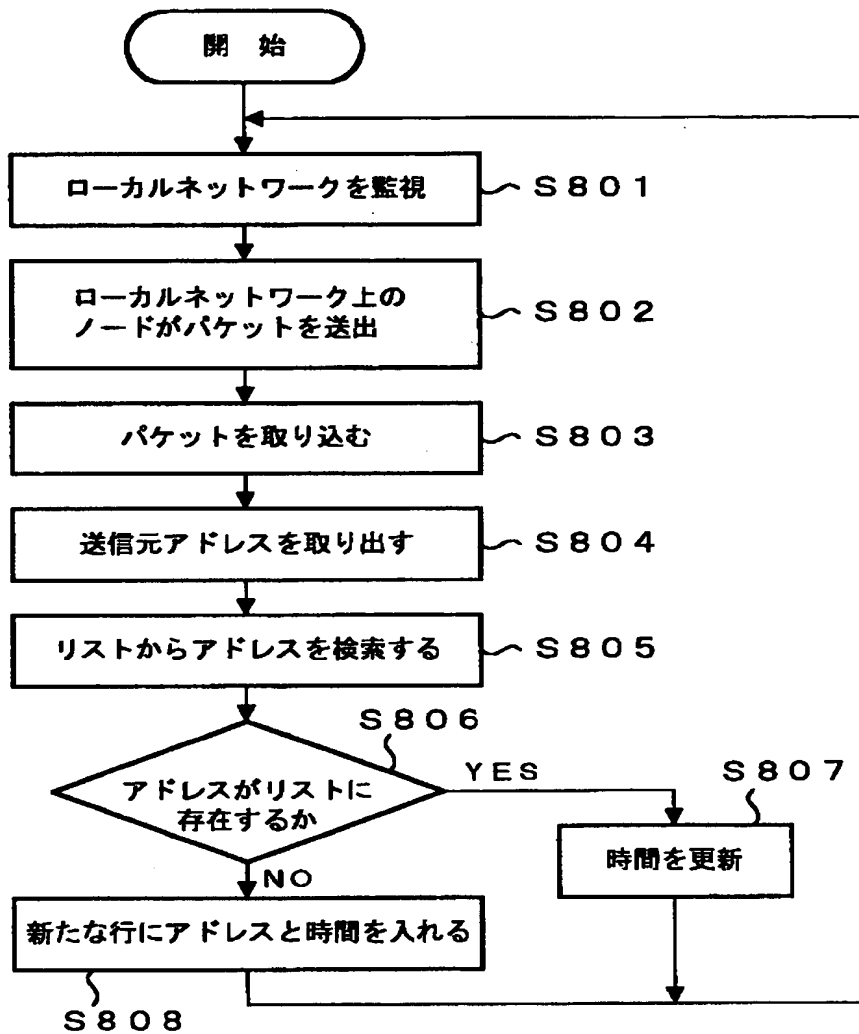
従来のフィードにおけるUDLR機能を説明するためのブロック図である。

【符号の説明】

1…フィード、2…ローカルI/F、3…UDL I/F、4…トンネルI/F、5…ローカルネットワーク、6…UDLネットワーク、7…トンネルネットワーク、17…レシーバ

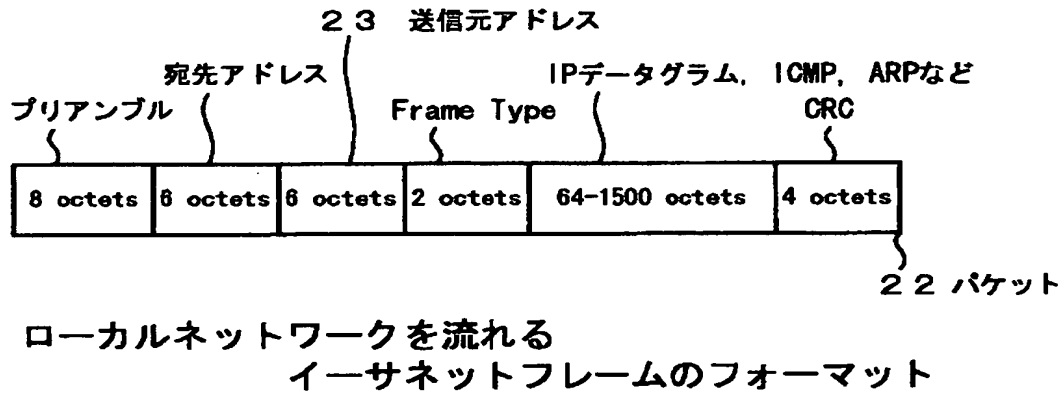
【書類名】 図面

【図 1】



アドレス学習の流れ

【図 2】



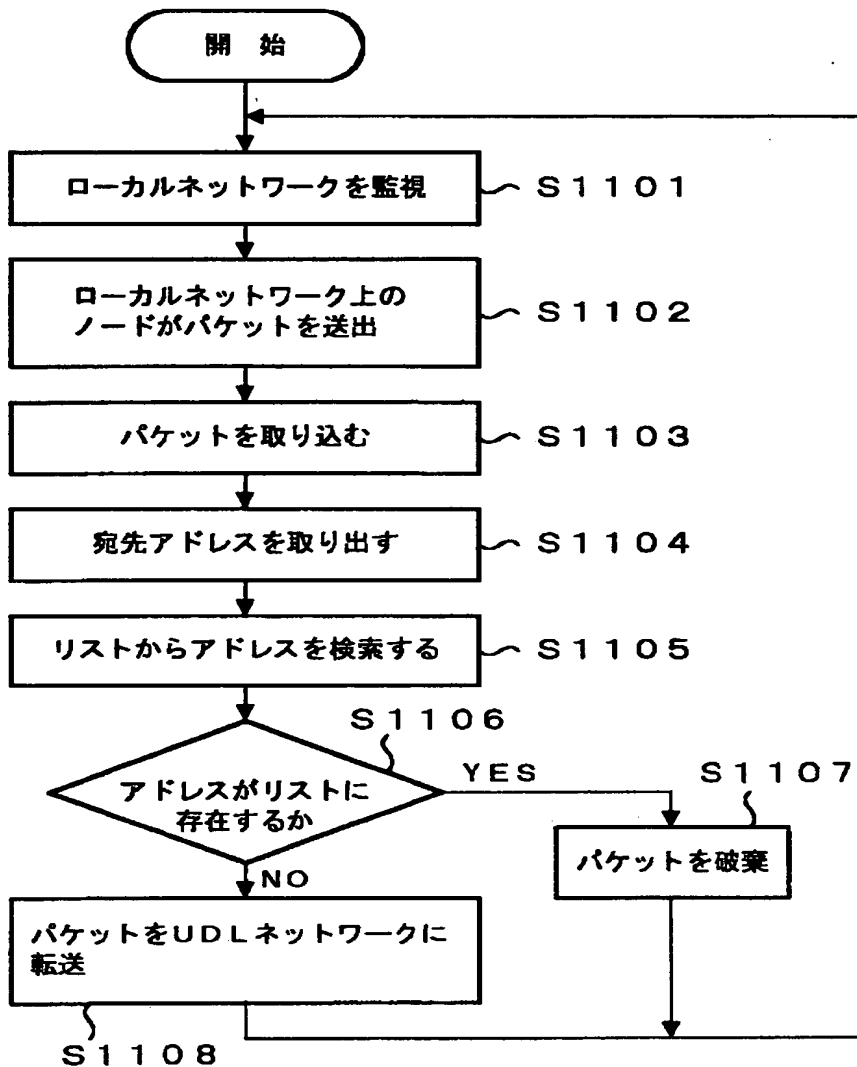
【図 3】

25 アドレスの列	26 更新時間の列
00:00:f8:1f:de:45	Fri Aug 7 23:31:02 1998
08:00:46:01:00:09	Wed Sep 9 15:00:23 1998
00:90:27:0c:46:98	Fei Feb 12 16:25:57 1999

24 リスト

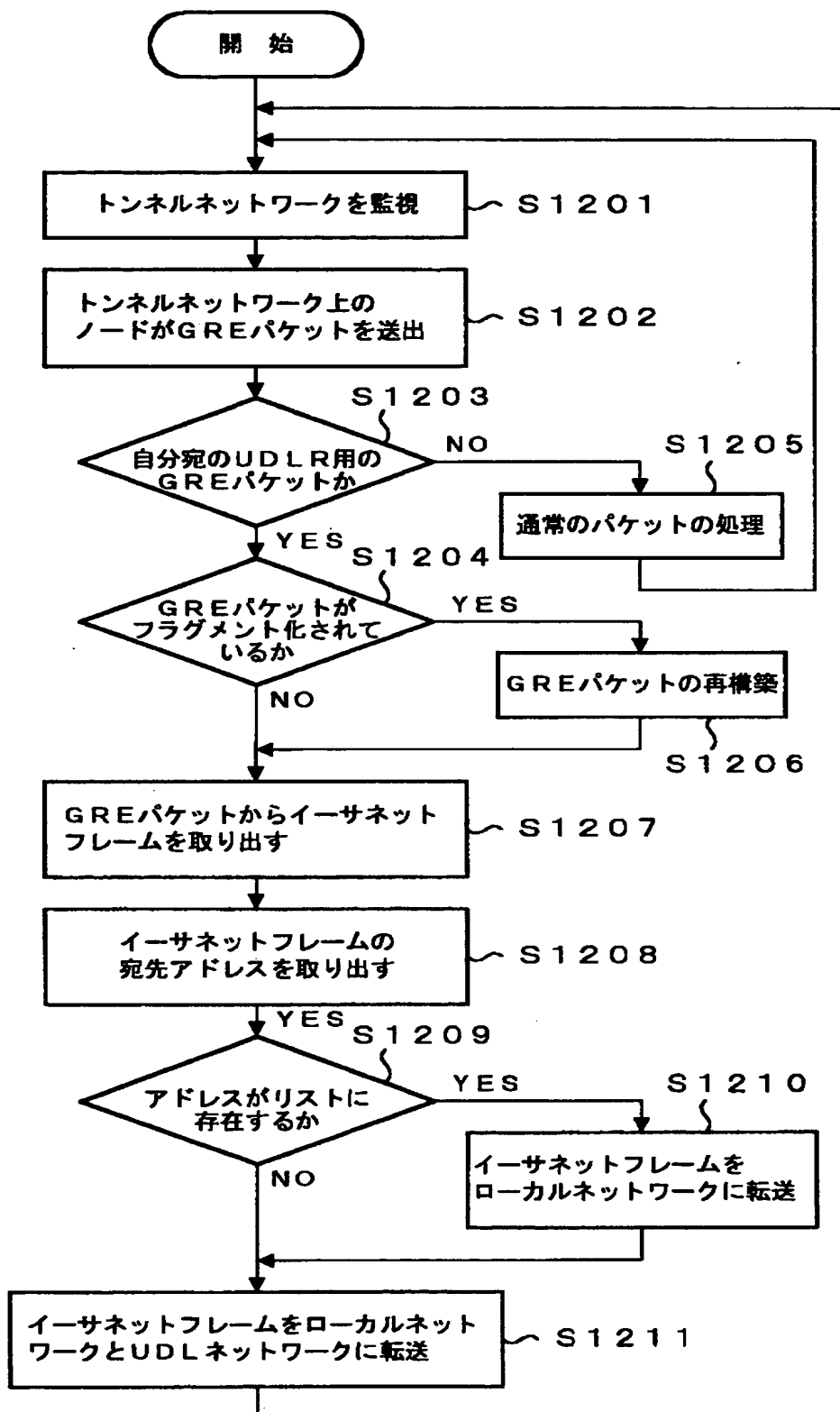
ローカルネットワーク上のノードが持つアドレスのリストの概念図

【図 4】



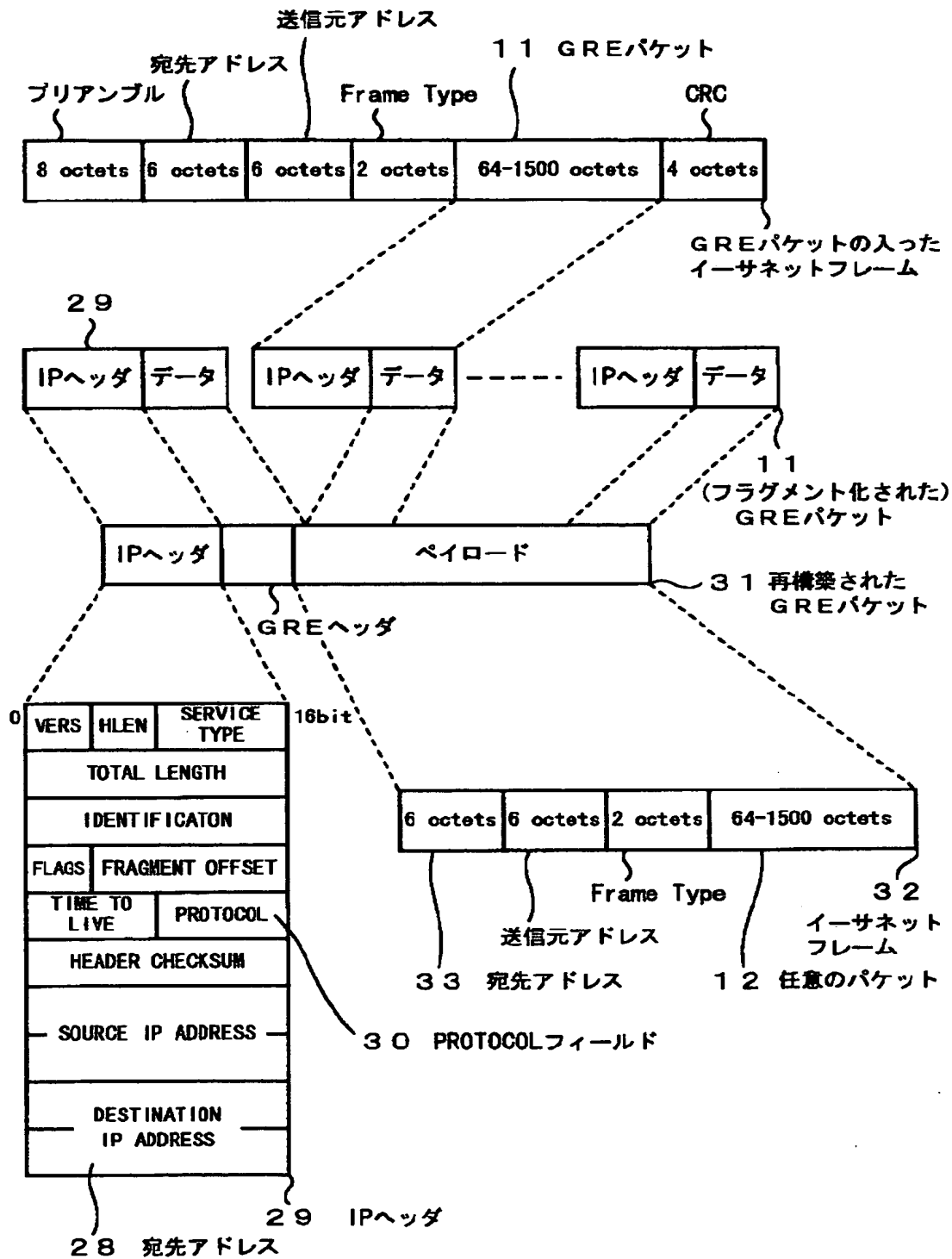
本発明によるパケット転送機能の流れ

【図 5】



本発明によるUDLR機能の流れ

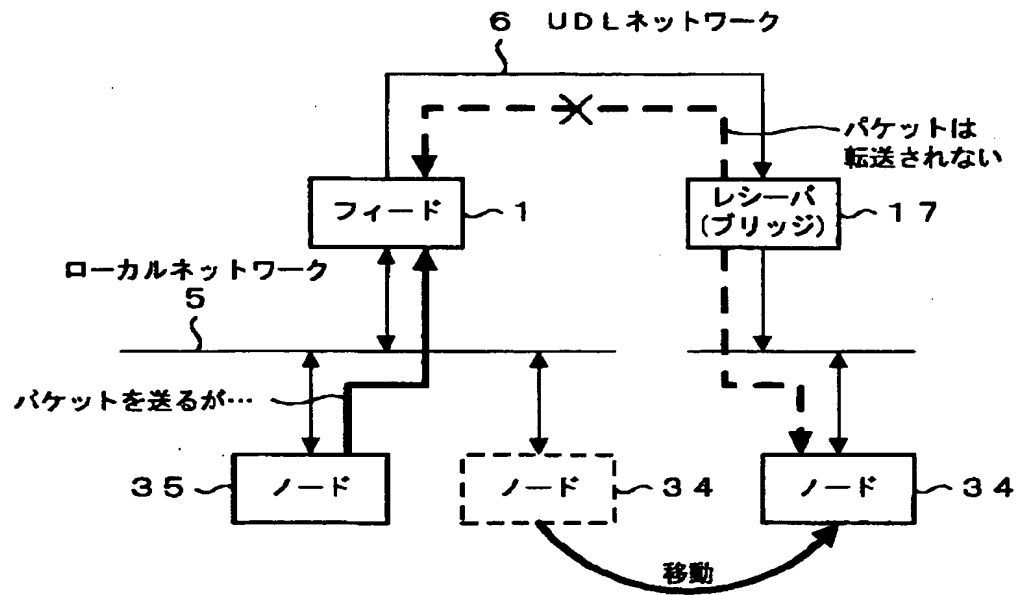
【図 6】



GRE パケットのフォーマット

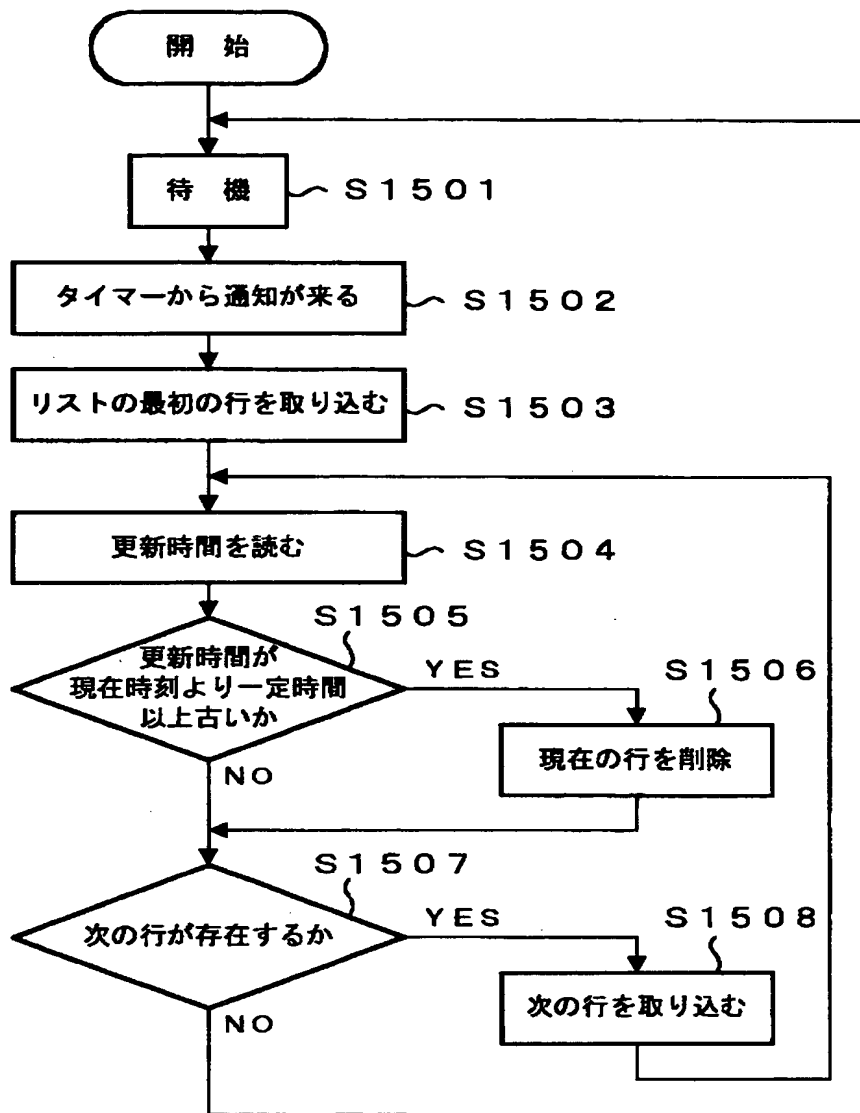


【図 7】



学習したアドレスを削除する必要がある場合

【図 8】



学習したアドレスの削除の流れ

【図 9】

38 有効／無効を示す例

25 アドレスの列

26 更新時間の列

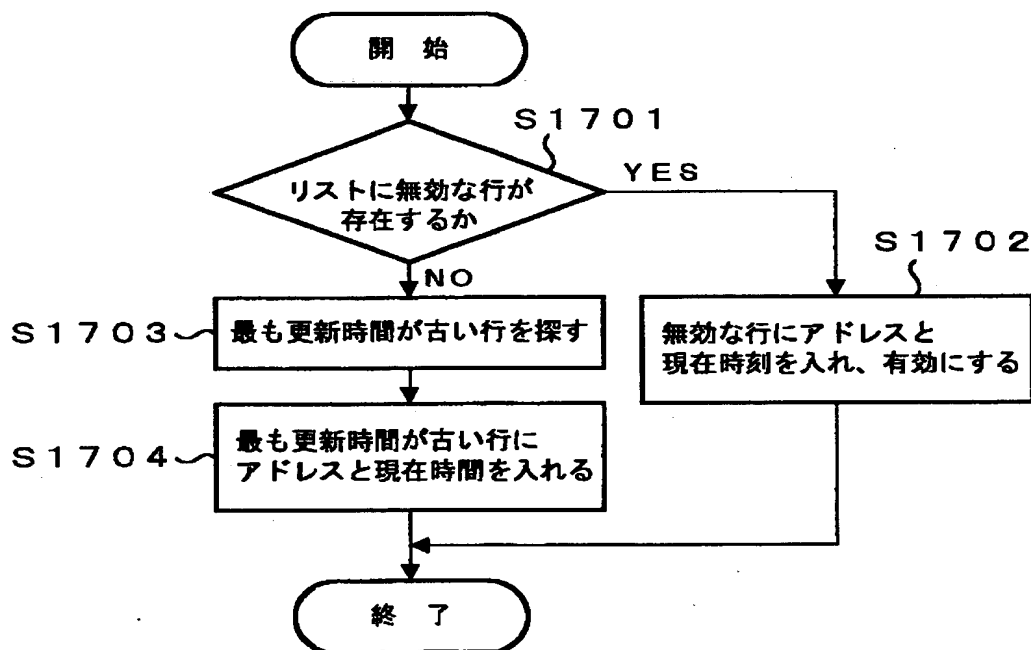
固定長

無効	00:00:f8:1f:de:45	Fri Aug 7 23:31:02 1998
有効	08:00:46:01:00:09	Wed Sep 9 15:00:23 1998
有効	00:90:27:0c:46:98	Fri Feb 12 16:25:57 1999
無効	(空)	(空)

37 リスト

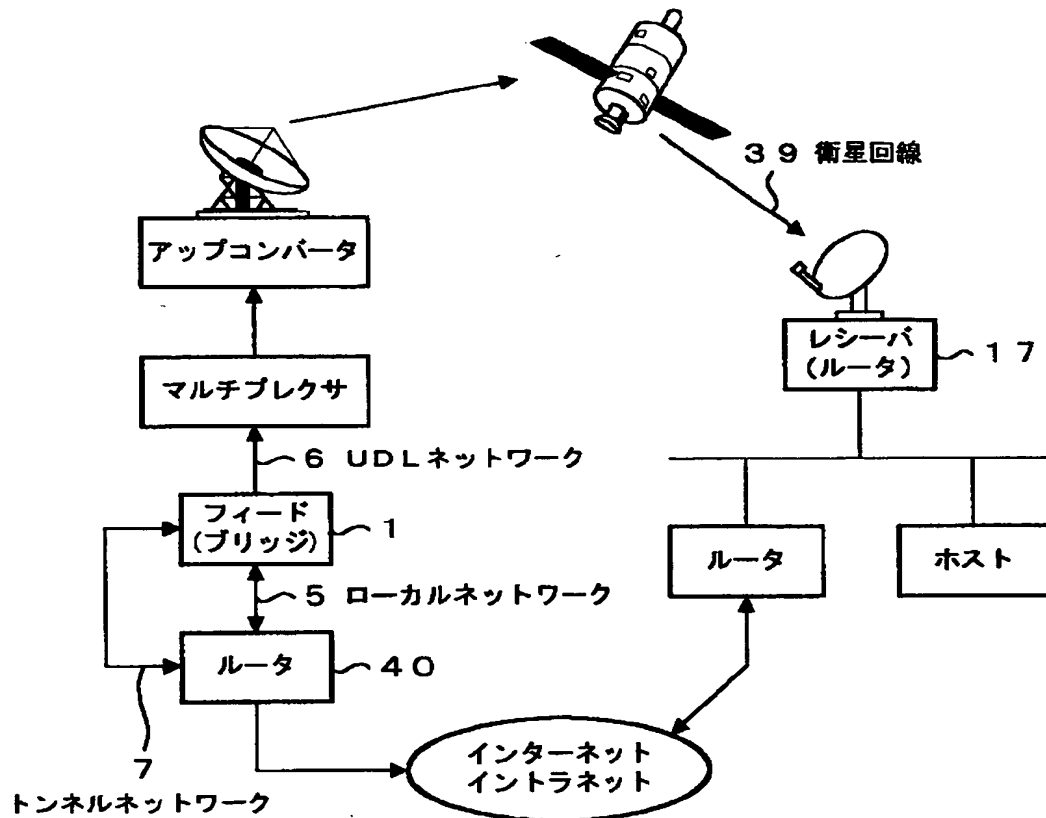
ローカルネットワーク上のノードが持つアドレスのリストの実装図

【図 10】



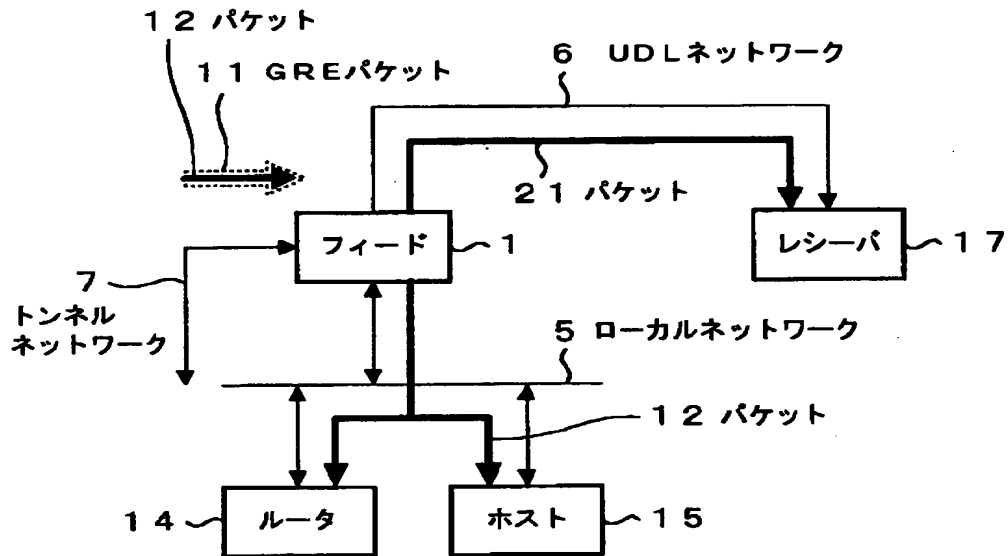
リストへの新しいアドレスの追加の流れ

【図 11】



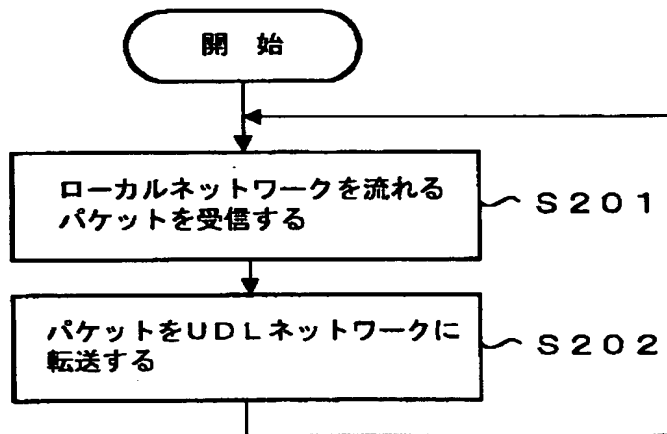
衛星回線とインターネット／イントラネットを  
用いたシステムへの適用例

【図 12】



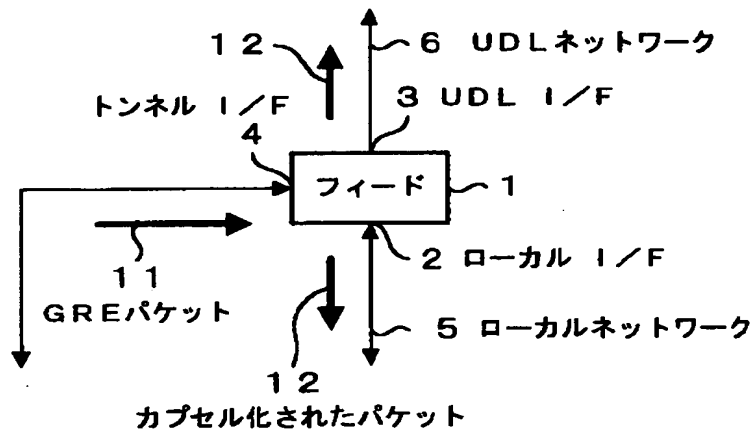
UDLRをサポートしたブリッジタイプのフィードの構成

【図 13】



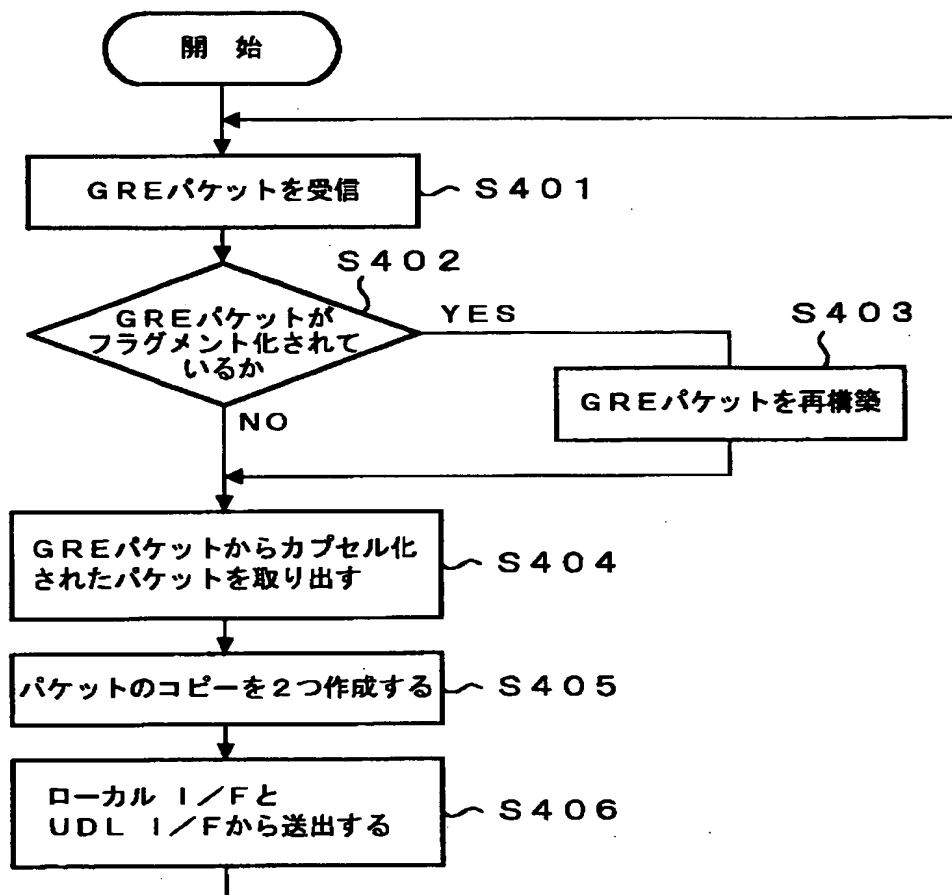
従来のパケット転送機能の流れ

【図 14】



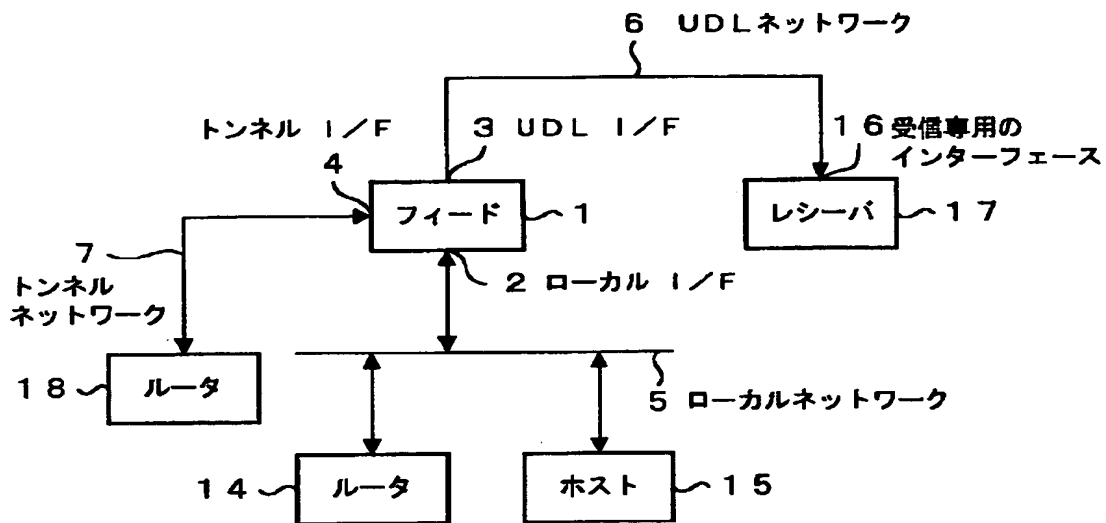
# UDLR機能

【図15】



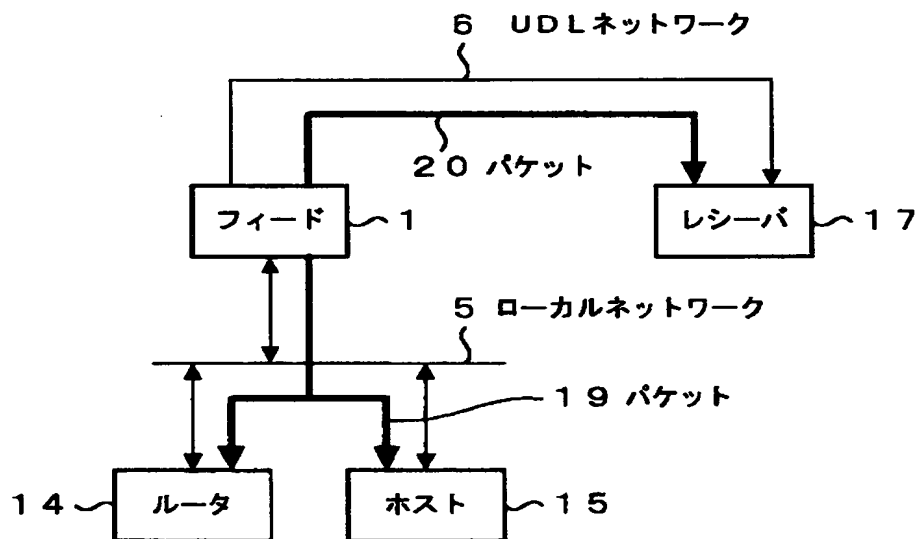
従来のUDLR機能の流れ

【図 16】



フィードの他の機器との接続形態

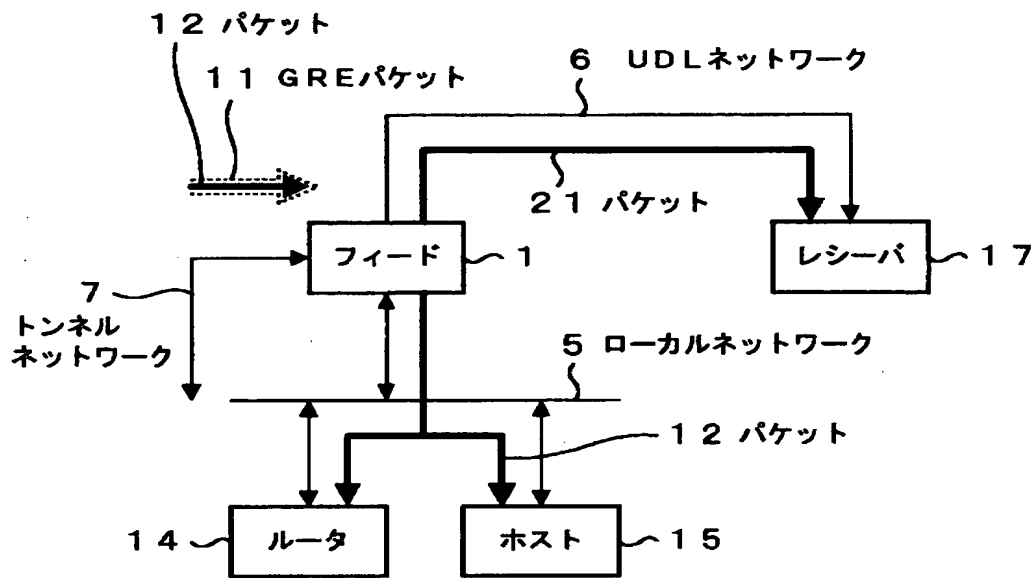
【図 17】



従来のフィードにおけるパケット転送機能の問題点



【図 18】



従来のフィードにおけるUDLR機能の問題点

【書類名】            要約書

【要約】

【課題】    片方向の通信回線を他の通信回線と組み合わせて、UDLRのように仮想的に双方向通信を行う場合に、不要なパケットを、フィードができるだけ生成しないようにし、UDLネットワークの帯域を有効に利用できるようにし、また、UDLネットワークに接続されたレシーバの様な機器の負荷を軽減する。

【解決手段】    フィードに所定のインターフェースから入力されたパケットの宛先を判断し、その判断したパケットの宛先からそのパケットかどのインターフェースの先にあるネットワークに送られるべきかを判断し、転送が必要な場合のみに所定のインターフェースから転送を行うようにした。

【選択図】            図 4

出 願 人 履 歴 情 報

識別番号 [000002185]

1. 変更年月日 1990年 8月30日

[変更理由] 新規登録

住 所 東京都品川区北品川6丁目7番35号

氏 名 ソニー株式会社